

03 Docker 安装与运行

更新时间: 2020-07-21 18:58:20



“知识犹如人体的血液一样宝贵。——高士其”

这一小节我们介绍一下 Docker 安装与运行，在开始之后，我们需要明白的时候，Docker 有两个版本：一个是商业版收费的：**Enterprise Edition**，一般简称 **Docker EE**；另一个是社区版，也就是 **Community Edition**，一般简称为 **Docker CE**。

这里安装我们会介绍在 **Linux** 平台、**Mac** 平台、**Windows** 平台的安装。重点会介绍一下在 **Linux** 平台的安装，**Linux** 平台又会细分两个：**Centos** 和 **Ubuntu**。

1. Linux 安装

Linux 下安装 Docker 我们将主要介绍 **Centos** 和 **Ubuntu** 系统

Centos

我们这里的示例 OS 版本是 **Centos 7.4**，使用命令 `lsb_release -a` 可以看到系统版本。

```
# lsb_release -a
LSB Version: :core-4.1-amd64:core-4.1-noarch
Distributor ID: CentOS
Description: CentOS Linux release 7.4.1708 (Core)
Release: 7.4.1708
Codename: Core
```

在 Centos 下面安装我们可以采用两种安装方式，一种是 **yum** 安装，一种是下载 **rpm** 包进行安装。

yum 安装

yum 安装需要确保 **yum** 源里面含有 **Docker** 软件包，国内的话推荐使用阿里云的 **yum** 源。如果要查看自己的系统的 **yum** 源，可以在目录 `/etc/yum.repos.d/` 下查看，下面是我自己的阿里云的 **ecs** 虚拟机 **yum** 源文件，包含两个部分：`CentOS-Base.repo` 和 `epel.repo`。内容分别如下

```
[base]
name=CentOS-$releasever
enabled=1
failovermethod=priority
baseurl=http://mirrors.cloud.aliyuncs.com/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=http://mirrors.cloud.aliyuncs.com/centos/RPM-GPG-KEY-CentOS-7

[updates]
name=CentOS-$releasever
enabled=1
failovermethod=priority
baseurl=http://mirrors.cloud.aliyuncs.com/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=http://mirrors.cloud.aliyuncs.com/centos/RPM-GPG-KEY-CentOS-7

[extras]
name=CentOS-$releasever
enabled=1
failovermethod=priority
baseurl=http://mirrors.cloud.aliyuncs.com/centos/$releasever/extras/$basearch/
gpgcheck=1
gpgkey=http://mirrors.cloud.aliyuncs.com/centos/RPM-GPG-KEY-CentOS-7
```

```
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
enabled=1
failovermethod=priority
baseurl=http://mirrors.cloud.aliyuncs.com/epel/7/$basearch
gpgcheck=0
gpgkey=http://mirrors.cloud.aliyuncs.com/epel/RPM-GPG-KEY-EPEL-7
```

使用阿里云的 **yum** 源安装 **Docker** 非常简单，强烈建议各位同学配置阿里云的 **yum** 源。安装使用下面一条命令即可：

```
yum install docker
```

执行完之后，可以通过如下命令查看 **Docker** 的版本：

```
# docker version
Client:
Version: 1.13.1
API version: 1.26
Package version:
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?
```

这个输出表示 **Docker** 的版本是 **1.13.1**。细心的同学会注意到最后一句提示，是因为 **Docker Daemon** 进程还没有启动，启动命令如下：

```
service docker start
```

或者：

```
systemctl start docker.service
```

启动完之后，我们就可以通过 `docker` 的命令来查看 `Docker` 是不是运行正常了，比如使用 `docker images` 查看本地所有的镜像。如下输出是正常的。

# docker images	TAG	IMAGE ID	CREATED	SIZE
-----------------	-----	----------	---------	------

下载 `rpm` 包安装

哪里下载呢？[当然是 dockerhub](#)，截图如下图所示：

Index of linux/centos/7/x86_64/stable/Packages/

```
../
containerd.io-1.2.0-1.2.beta.2.el7.x86_64.rpm 2019-10-18 21:56:46 22.6 MiB
containerd.io-1.2.0-2.0.rc.0.1.el7.x86_64.rpm 2019-10-18 21:56:46 22.1 MiB
containerd.io-1.2.0-2.2.rc.2.1.el7.x86_64.rpm 2019-10-18 21:56:46 22.1 MiB
containerd.io-1.2.0-3.el7.x86_64.rpm 2019-10-18 21:56:46 22.1 MiB
containerd.io-1.2.10-3.2.el7.x86_64.rpm 2019-10-18 21:56:46 23.1 MiB
containerd.io-1.2.2-3.3.el7.x86_64.rpm 2019-10-18 21:56:47 22.1 MiB
containerd.io-1.2.2-3.el7.x86_64.rpm 2019-10-18 21:56:47 22.1 MiB
containerd.io-1.2.4-3.1.el7.x86_64.rpm 2019-10-18 21:56:47 22.1 MiB
containerd.io-1.2.5-3.1.el7.x86_64.rpm 2019-10-18 21:56:47 22.1 MiB
containerd.io-1.2.6-3.3.el7.x86_64.rpm 2019-10-18 21:56:47 25.9 MiB
docker-ce-17.03.0.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:47 18.6 MiB
docker-ce-17.03.1.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:47 18.6 MiB
docker-ce-17.03.2.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:47 18.6 MiB
docker-ce-17.03.3.ce-1.el7.x86_64.rpm 2019-10-18 21:56:47 18.6 MiB
docker-ce-17.06.0.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:47 20.6 MiB
docker-ce-17.06.1.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:48 20.6 MiB
docker-ce-17.06.2.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:48 20.6 MiB
docker-ce-17.09.0.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:48 21.1 MiB
docker-ce-17.09.1.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:48 21.1 MiB
docker-ce-17.12.0.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:48 30.4 MiB
docker-ce-17.12.1.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:49 30.4 MiB
docker-ce-18.03.0.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:49 34.6 MiB
docker-ce-18.03.1.ce-1.el7.centos.x86_64.rpm 2019-10-18 21:56:49 34.6 MiB
docker-ce-18.06.0.ce-3.el7.x86_64.rpm 2019-10-18 21:56:49 40.7 MiB
docker-ce-18.06.1.ce-3.el7.x86_64.rpm 2019-10-18 21:56:49 40.7 MiB
docker-ce-18.06.2.ce-3.el7.x86_64.rpm 2019-10-18 21:56:50 40.7 MiB
docker-ce-18.06.3.ce-3.el7.x86_64.rpm 2019-10-18 21:56:50 40.7 MiB
docker-ce-18.09.0-3.el7.x86_64.rpm 2019-10-18 21:56:50 18.7 MiB
docker-ce-18.09.1-3.el7.x86_64.rpm 2019-10-18 21:56:50 18.7 MiB
docker-ce-18.09.2-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.3-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.4-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.5-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.6-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.7-3.el7.x86_64.rpm 2019-10-18 21:56:51 18.7 MiB
docker-ce-18.09.8-3.el7.x86_64.rpm 2019-10-18 21:56:52 18.7 MiB
docker-ce-18.09.9-3.el7.x86_64.rpm 2019-10-18 21:56:52 21.2 MiB
docker-ce-19.03.0-3.el7.x86_64.rpm 2019-10-18 21:56:52 24.3 MiB
docker-ce-19.03.1-3.el7.x86_64.rpm 2019-10-18 21:56:52 24.3 MiB
```

其中 `containerd` 和 `docker-ce-cli` 是 `docker-ce` 安装的前置条件，也就是要安装 `docker-ce`，需要先安装 `containerd` 和 `docker-ce-cli`。版本之间保证 `docker-ce` 和 `docker-ce-cli` 的版本一致，`containerd` 的版本尽可能新就 `ok`。下载到本地之后，使用 `yum` 安装。

```
yum install <pkg_name>.rpm
```

我这里选的版本是：

- `docker-ce-cli-19.03.5-3.el7.x86_64.rpm`
- `containerd.io-1.2.6-3.3.el7.x86_64.rpm`
- `docker-ce-19.03.5-3.el7.x86_64.rpm`

按上面的顺序安装之后启动 docker

```
systemctl start docker.service
```

我们看一下 docker version:

```
# docker version
Client: Docker Engine - Community
Version: 19.03.5
API version: 1.40
Go version: go1.12.12
Git commit: 633a0ea
Built: Wed Nov 13 07:25:41 2019
OS/Arch: linux/amd64
Experimental: false

Server: Docker Engine - Community
Engine:
Version: 19.03.5
API version: 1.40 (minimum version 1.12)
Go version: go1.12.12
Git commit: 633a0ea
Built: Wed Nov 13 07:24:18 2019
OS/Arch: linux/amd64
Experimental: false
containerd:
Version: 1.2.6
GitCommit: 894b81a4b802e4eb2a91d1ce216b8817763c29fb
runc:
Version: 1.0.0-rc8
GitCommit: 425e105d5a03fabd737a126ad93d62a9eeede87f
docker-init:
Version: 0.18.0
GitCommit: fec3683
```

这个要比阿里云的 Yum 源中的 Docker 版本高很多，如果需要新的版本的 feature 的话，建议自己下载对应的 rpm 包进行安装。

Ubuntu

我们知道 ubuntu 的包管理是通过 apt-get 来做的，但是默认的 apt-get 里面是没有包含 docker 的软件包的，我们需要将 Docker 的官方仓库加进来。

apt-get 安装

更新 ubuntu 的 apt 源索引

```
sudo apt-get update
```

安装包允许apt通过HTTPS使用仓库

```
sudo apt-get install \
apt-transport-https \
ca-certificates \
curl \
software-properties-common
```

添加Docker官方GPG key

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

设置Docker稳定版仓库

```
sudo add-apt-repository \
"deb [arch=amd64] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) \
stable"
```

添加仓库后，更新apt源索引

```
sudo apt-get update
```

安装 Docker CE，不输入 Version，则默认安装最新版本

```
sudo apt-get install docker-ce=<VERSION>
```

启动 Docker

```
systemctl start docker.service
```

使用这种方式安装的最大劣势就是 Docker 官方仓库和我们国内的机器网速捉急，相比之下将 deb 包下载到本地再进行安装往往是一种更好的安装方式。

deb 软件包安装

软件包下载地址为：<https://download.docker.com/linux/ubuntu/dists/>，这个地方先选择自己的 ubuntu 版本，比如我们选择 artful，则软件包的完整路径为：<https://download.docker.com/linux/ubuntu/dists/artful/pool/stable/amd64/>

Index of linux/ubuntu/dists/artful/pool/stable/amd64/

.. /	
docker-ce_17.12.0-ce-0ubuntu_amd64.deb	2018-06-08 17:59:05 28.4 MiB
docker-ce_17.12.1-ce-0ubuntu_amd64.deb	2018-06-08 17:59:05 28.4 MiB
docker-ce_18.03.0-ce-0ubuntu_amd64.deb	2018-06-08 17:59:05 32.3 MiB
docker-ce_18.03.1-ce-0ubuntu_amd64.deb	2018-06-08 17:59:06 32.3 MiB
docker-ce_18.06.0-ce-3-0ubuntu_amd64.deb	2018-07-18 22:51:33 38.3 MiB
docker-ce_18.06.1-ce-3-0ubuntu_amd64.deb	2018-08-21 23:04:08 38.4 MiB
docker-ce_18.06.2-ce-3-0ubuntu_amd64.deb	2019-02-11 18:11:11 38.3 MiB
docker-ce_18.06.3-ce-3-0ubuntu_amd64.deb	2019-02-20 17:34:21 38.3 MiB

然后我们选择一个 docker-ce 版本，直接本地安装即可。

```
sudo dpkg -i <deb pkg name>
```

2. Mac 安装

Mac 版本的安装页面地址为：<https://docs.docker.com/v17.09/docker-for-mac/install/#download-docker-for-mac>。

Docker 针对 Mac 平台提供了两个不同的版本：

- stable: 稳定版
- edge: 最新的 release，稳定性不保障

Stable channel

This installer is fully baked and tested. This is the best channel to use if you want a reliable platform to work with. These releases follow the Docker Engine stable releases.

On this channel, you can select whether to send usage statistics and other data.

Stable builds are released once per quarter.

[Get Docker for Mac \(Stable\)](#)

Edge channel

This installer provides the latest Edge release of Docker for Mac and Engine, and typically offers new features in development. Use this channel if you want to get experimental features faster, and can weather some instability and bugs. We collect all usage data on Edge releases across the board.

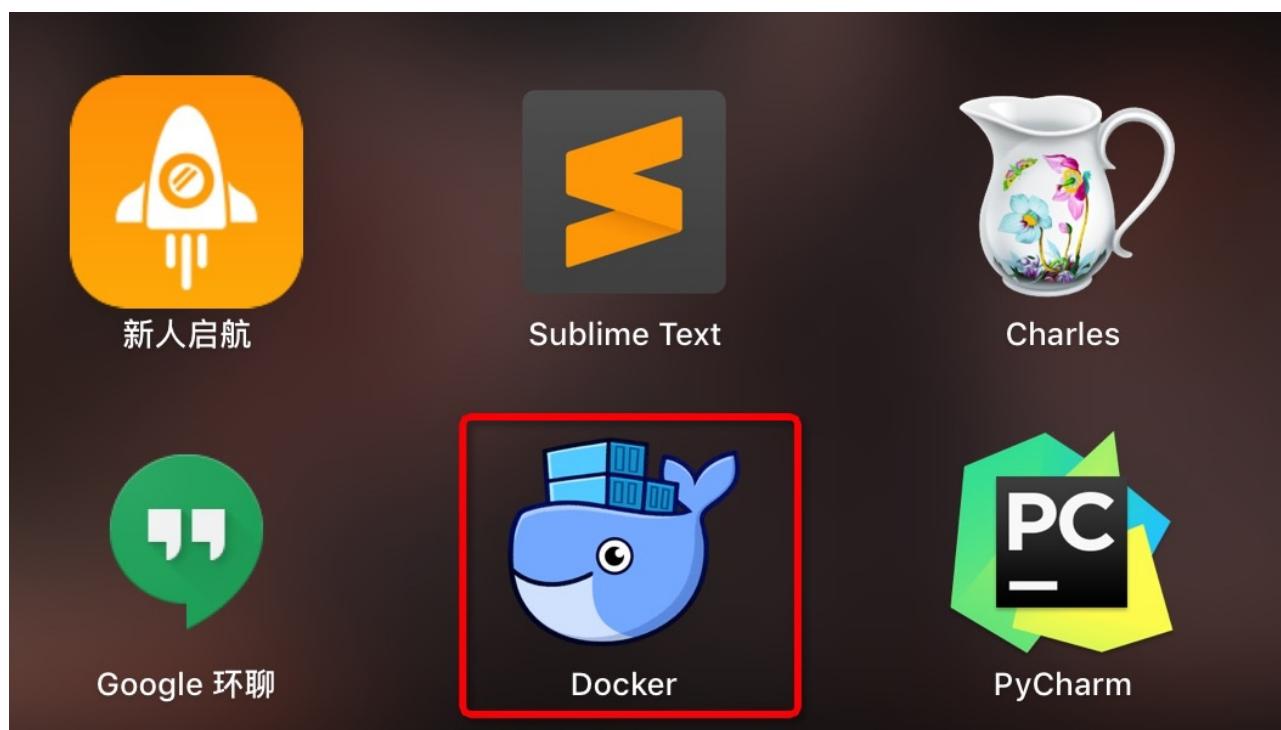
Edge builds are released once per month.

[Get Docker for Mac \(Edge\)](#)

然后我们通过上图中“Get Docker for Mac”链接获取 **dmg** 安装文件。

启动 Docker

启动，找到你的 Application 中的下面这个 Docker 图标，点击运行即可。



启动之后，我们可以在导航栏上发现这么一个小图标，就表示 Docker 运行成功了。



3. Windows 安装

Windows 安装链接: <https://docs.docker.com/v17.09/docker-for-windows/install/#download-docker-for-windows> , 和 Mac 版类似, 提供了 stable 和 edge 版本。我们只需要下载对应的版本的可执行文件进行安装即可。启动也是类似的, 这里就不再赘述了。

}

← 02 成王败寇: 容器的发展之路

04 小身材大能量: Docker 概览 →